AMENDMENTS TO THE SPECIFICATIONS

Please replace the following paragraphs.

[0021] The location and perspective from which a user may view the scene may be defined by what is referred to as a scene camera and may also be referred to as a camera view. In one embodiment, multiple scene cameras for a scene may be supported such that a user may switch back and forth between two, and in other embodiments, more, scene cameras. In various embodiments, there may be multiple and varied supported 2D and 3D objects. In various embodiments, multiple formats of 2D and 3D objects may be supported including, for example, OpenGL3D. An example of a 2D media object is a bitmap image which, in one embodiment, may be known as a sprite. In one embodiment, the source for a sprite can be a single bitmap, a bitmap image sequence, a vector image, a video track, a live stream, or a source specified by a uniform resource locator (URL). Examples of 3D media objects include object movies and OpenGL 3D objects. Object movies are not truly movies and are not truly 3D, but they are an enhanced form of 2D having some 3D properties. Although not truly 3D entities, object movies and other similarly enhanced 2D objects will be referred to as 3D objects and as enhanced 2D objects herein. Object movies are created by photographing or rendering a series of views of the subject at carefully spaced angles of tilt and rotation and assembling the images. In one embodiment, each view of the subject in an object movie is a still image. In another embodiment, one or more views of the subject of an object movie may be a video clip or animation. In one embodiment, an object movie may be created by photographing a series of images of an object as the object is rotated on a platter along different axes. In another embodiment, the series of images comprising an object movie may be captured when rotating a camera from around the object along different axes from a fixed distance.

[0026] Generally, tilting a scene or changing a user's view of a scene up or down, requires that the scene be rotated about the scene's x-axis, as shown by reference number 320. When a scene is to be tilted, a function or tool provided by a prior art library of APIs is called to rotate each of the objects based on the translation vector and

the amount of the rotation along the scene's x-axis. In one embodiment, this may be accomplished by retrieving an appropriate image frame of the object at a corresponding orientation. Similarly, in some embodiments, support for yaw may also be provided. Changing the yaw of a scene, or changing a user's view in relation to a vertical axis, requires that the scene be rotated about the scene's ~~y-axis~~ z-axis, as shown by reference number 330. To rotate a scene, a function or tool provided by a prior art library of APIs is called to rotate each of the scene objects based on their translation vectors and the user requested rotation of the scene about the scene's z-axis. Depending on the object and the embodiment, the rotation may be achieved by application of a software algorithm or by retrieving an appropriate image frame captured corresponding to the desired orientation.

[0029] **Figure 4B** illustrates an object coordinate system superimposed on a scene coordinate system. Each object in a scene is defined as a series or sequence of image frames and orientation data. These image frames may be retrieved, accessed and manipulated by available library functions accessing the media object. Each object is initially defined in its own coordinate system. However, when creating a scene, a scene coordinate system is created. To exist in a scene, each object added to the scene must be placed at a location in the scene. The definition of where each object is to exist in a scene is determined by a translation vector which is used in conjunction with the object's coordinate system. For example, each object's position and orientation may be defined by a 3x3 matrix representing object axes 410, 412 and 414, as shown in **Figure 4B** and a translation vector. The translation vector directs where to place the object in relation to scene axes ~~422, 424 and 426~~ 420, 422 and 424, as shown in **Figure 4B**. The 3x3 rotation matrix is combined with the translation vector to form a 4x4 matrix known as a scene object.